0

UNLIMITED DISTRIBUTION

National Defence
Research and Development Branch

Défense Nationale
Bureau de Recherche et Développment

TECHNICAL COMMUNICATION 88/303
October 1988

# MOPRE:
# A PRE-PROCESSOR FOR DEFINING AND MODIFYING INPUT TO THE SEAKEEPING ANALYSIS PROGRAM SHIPMO

J.L. Colwell - D.J. Colter
F. Lamanque

Defence
Research
Establishment
Atlantic

Centre de
Recherches pour la
Défense
Atlantique

DEFENSIO PER SCIENTIAM

Canada

UNLIMITED DISTRIBUTION

National Defence
Research and Development Branch

Défense nationale
Bureau de recherche et développement

# MOPRE:
# A PRE-PROCESSOR FOR DEFINING AND MODIFYING INPUT TO THE SEAKEEPING ANALYSIS PROGRAM SHIPMO

J.L. Colwell - D.J. Colter
F. Lamanque

October 1988

Approved by W.C.E. Nethercote H/Hydronautics Section

DISTRIBUTION APPROVED BY

D/TD

TECHNICAL COMMUNICATION 88/303

Defence
Research
Establishment
Atlantic

Centre de
Recherches pour la
Défense
Atlantique

Canada

## Abstract

The FORTRAN program MOPRE is an interactive, user-friendly pre-processor for creating, modifying and validating input files for the "strip-theory" seakeeping analysis program SHIPMO. An overview introduces the commands and general procedures used in MOPRE to produce the SHIPMO input and defines the record/sub-record structure used by both programs. The MOPRE commands are described in detail, with emphasis on procedures which simplify defining and modifying ship geometry. The general features of the MOPRE interface are described, including: on-line help, abbreviation recognition, error detection, and a Command-Line Editor. Computer requirements are discussed and an optional, operating system-dependent menu-system for controlling execution of MOPRE, SHIPMO and other seakeeping programs is described.

## RÉSUMÉ

Le programme MOPRE en FORTRAN est un préprocesseur interactif facile à utiliser et sert à créer, modifier et valider les fichiers d'entrée dans le programme d'analyse "par segmentation" du comportement des navires en mer SHIPMO. L'aperçu contient les instructions et procédures générales utilisées dans MOPRE pour générer les entrées dans SHIPMO et détermine la structure des enregistrements et sous-enregistrements dans les deux programmes. Les instructions de MOPRE sont décrites en détail et traitent surtout des procédures destinées à simplifier les tâches de définition et de modification de la géométrie des navires. Les charactéristiques générales de l'interface de MOPRE y sont décrites et comprennent notamment l'aide en direct, la reconnaissance des abréviations, la détection des erreurs et un éditeur de ligne de commande. L'aperçu traite des besoins informatiques et contient une description d'un ensemble facultatif de menus asservis au système d'exploitation servant à contrôler l'exécution des programmes MOPRE, SHIPMO et d'autres programmes d'analyse du comportement des navires en mer.

# Contents

# 1 INTRODUCTION

MOPRE is an interactive program for defining and modifying input to SHIPMO, the DREA "strip-theory" seakeeping analysis computer program documented in References 1 through 5. MOPRE is compatible with the most recent version of SHIPMO, called SHIPMO4 [5][1], but files from the older versions of SHIPMO described in References 1 and 4 can be input to MOPRE. MOPRE is not compatible with the versions of SHIPMO which consider wave loads, as described in References 2 and 3. This Technical Memorandum is a user's guide for MOPRE; it does not replace the SHIPMO user's guide presented in Reference 5.

SHIPMO input includes a variety of parameters which define a title, sea state(s), ship speed(s), hull form geometry, and details of appendages. These parameters are arranged sequentially in a record/sub-record format and are stored in a disk file for input to SHIPMO. The size of this input file varies from a minimum of approximately 110 individual numbers to a maximum of over 1100 numbers. The typical input file consists of approximately 750 numbers. Since the SHIPMO input file is arranged in an unlabelled format, defining and especially modifying an input file is laborious. Also, the inter-dependence of records with sub-records and with other records means that changing a single number often requires significant changes in other parts of the file. Moreover, it is not possible to verify whether an input file is acceptable before executing SHIPMO. When SHIPMO encounters insufficient or incorrectly arranged input, execution is aborted and the user must locate the error or errors with no help other than the required structure as defined in the user's manual.

Program MOPRE combines command- and menu-driven procedures with an on-line help facility to provide a robust, user-friendly method for creating, modifying and validating SHIPMO input files. The command-driven segments of the MOPRE interface provide abbreviation recognition, error detection and command completion. All requests for user-supplied input are preceeded by an informative prompt which describes the parameter, defines its name, and when applicable, defines the units expected for input.

This document describes MOPRE in three stages. First, an overview of MOPRE summarizes the commands and procedures for creating and modifying SHIPMO input files, and defines the record/sub-record structure used by both MOPRE and SHIPMO. Second, detailed descriptions of the MOPRE commands and procedures are presented. Finally, the general properties of the MOPRE user-interface are described and the computing requirements for MOPRE are discussed. The appendices contain examples of using MOPRE to create a new input file and to modify an existing file.

The overview of MOPRE presented in Section 2 of this document and the on-line help available when running MOPRE provide sufficient information to create and modify SHIPMO input files effectively. It is recommended that new users should first read the overview of MOPRE, and then use the program to become familiar with its basic procedures and commands. Most of the information in Sections 3 to 5 of this document is provided as a source reference on advanced command procedures and on implementing program MOPRE.

[1] numerals within [...] brackets denote references at the end of this document.

## 2 Overview of MOPRE

MOPRE has three functions with respect to SHIPMO input files: creating new files; retrieving library files; and, modifying existing files. The MOPRE commands introduced in this section are discussed in more detail in Section 3.

The function of creating new files is relatively simple; the command CREATE activates a series of prompts which request input for all SHIPMO records and sub-records. The record/sub-record structure is defined later in this section. Once a file is completely defined, the information is stored using the SAVE command, which creates two disk files; one is reserved for actual input to SHIPMO, and the other is a library file. The name of the library file is defined by the user and the name of the reserved input file is automatically set to the operating system default for FORTRAN (FOR002.DAT for VAX VMS).

It is possible to suspend the CREATE mode before all input is defined by entering "Z" at any prompt[2]. The two most common reasons for suspending the CREATE mode are to correct an error using the CHANGE command or to SAVE the existing information and EXIT from MOPRE. In the first case, the CHANGE command activates procedures which provide access to all input parameters, as described in Section 3. In the second case, when the SAVE command is used on an incomplete file (i.e. not all required input parameters have been defined), then only the library file is produced (i.e. FOR002.DAT is only produced for completely defined input). In either case, the CREATE mode is re-activated using the command CONTINUE_CREATING[3]. When the user wants to CONTINUE_CREATING a file which was saved in a previous session with MOPRE, then the ENTER command is first used to load the file into MOPRE.

The other MOPRE functions of retrieving library files and modifying existing (library) files are based on the commands introduced above. To retrieve a library file for execution with SHIPMO, simply ENTER the file, SAVE it, EXIT from MOPRE and RUN SHIPMO. When the name of a desired library file is not known, then the FIND command displays the titles and names of all library files. The title is simply an input record which describes the contents of a file and which is printed at the top of SHIPMO lineprinter output, as described in Reference 5. A similar procedure is used to modify existing files, except the CHANGE command is used to alter the original file before saving it. The SHOW command displays the current value of an individual parameter, all parameters within a particular record or sub-record, or the current values of all parameters.

The HELP command invokes the MOPRE help facility which describes all commands and procedures. Also, entering "?" at any prompt provides information on the input expected at that prompt. For example, when ? is entered with no preceeding command, then the list of acceptable MOPRE commands is displayed. Alternately, when ? is entered at a prompt expecting input for an individual parameter, then any relevant information on that parameter is displayed.

[2] Z is a convenient synonym for EXIT or QUIT, all of which are recognized by MOPRE.

[3] the "_" symbol is used in lieu of a space for multiple-word names and phrases which are treated as a single word by the MOPRE command interface.

The Appendices contain two examples of using MOPRE: Appendix A shows generating a new file using the CREATE command and Appendix B shows modifying an existing file using the CHANGE command. In the second example, two changes are made to the original file. First, the sea state description is changed from input of measured spectral data to the Bretschneider two parameter spectrum and, second, the values of a few offset points are changed. Changing the sea state definition requires significant changes to the actual input file: four of the original input records are modified, one new record is defined, and two original records are omitted; however, using MOPRE simplifies this process, as shown in Appendix B.

## 2.1 Record/Sub-Record Structure of Input

The SHIPMO user's manual [5] defines the input record structure used in both SHIPMO and MOPRE; however, there are some differences in the way this structure is implemented by MOPRE. The primary difference is that SHIPMO uses alphabetical labels for each record, while MOPRE uses descriptive names, as shown below.

| INPUT RECORDS | |
|---|---|
| MOPRE Record Name | SHIPMO Record Label |
| TITLE | (a) |
| CONTROL INTEGERS | (b) |
| WAVE FREQUENCIES | (c) |
| ENCOUNTER FREQUENCIES | (d) (d1) |
| SEA STATE | (e) (e1) (f) (f1) (f2) (f3) |
| SHIP SPEEDS | (g) |
| HULL FORM | |
| BASIC SHIP DATA | (h) |
| STATION DEFINITION | (i) (i1) |
| OFFSETS | (i2) (i3) |
| HULL COEFFICIENTS | (i2a) |
| LOAD WATERLINE | (i4) (i5) (i6) |
| SEAKEEPING and SLAMMING | (j) (j1) (j2) (j2a) |
| APPENDAGES | |
| BILGE KEELS | (k) (k1) (k2) |
| SKEG | (l) |
| BRACKETS | (m) (m1) (m2) (m5) |
| RUDDER | (m) (m3) (m4) |
| ANTI-ROLLING TANK | (n) |

The only other significant differences between the MOPRE and SHIPMO record structure are that MOPRE introduces HULL FORM and APPENDAGES as records and delegates the related items to sub-record status, and the SHIPMO Record (o) is not defined in MOPRE. This omitted record, called "Record (o), Control Integer for Next Case" in the

SHIPMO users manual [5], is used to define when another input file is appended directly to the one just read-in by SHIPMO. When using MOPRE, the command BATCH (described in the next section) provides a different method for defining a multiple-input-file execution of SHIPMO.

# 3 MOPRE Commands

The complete set of MOPRE commands is displayed at the terminal as follows.

```
MOPRE Command => ?

 Commands are:

BATCH        CHANGE       CREATE       CONTINUE_CREATING        DEFINE
DELETE       ENTER        EXIT         FIND         HELP        SAVE
SET          SHOW

 or.....a record or subrecord name (SHOW RECORD ? for listing)
```

These commands are described in the following sections. Note that this information is provided as a source reference on the details of command procedures and options. The overview of MOPRE in the previous section and the MOPRE on-line help facility provide sufficient information to create and modify input files for SHIPMO effectively. It is recommended that the new user should run MOPRE to become familiar with its basic structure and procedures before absorbing the detailed information provided below.

## 3.1 BATCH

The BATCH command produces an operating system file which controls batch execution of SHIPMO using one or more input files. The user defines the name of each file and then MOPRE writes the control file to disk.

Currently, MOPRE only produces batch control files for the VAX VMS operating system [7], but the algorithm can be easily modified for other systems by editing the MOPRE source. The program-control environment described in Section 5.4 includes an option for submitting this batch control file for execution on a VAX VMS system. If this environment is not available, then the same batch control file can be submitted using operating system commands.

## 3.2 CHANGE

The CHANGE command alters the current value of a parameter, but it can also define a parameter's initial value, as discussed in the next section on the CREATE command.

This command requires two words of input, the first being CHANGE (abbreviation CH) and the second is the name of the record to be changed. The names of records are displayed when the user enters ? as part of the CHANGE command (e.g. ch?) or in response to the prompt for insufficient input (e.g. CHANGE ? => ?). In one case, the record name defines an individual parameter (i.e. title), but it usually defines a collection of parameters which are accessed through a menu. The special properties of CHANGE OFFSETS are discussed later.

Most menus accessed by CHANGE allow the user to select an individual parameter or ALL. When ALL is selected, the user is sequentially prompted to provide input for each parameter in that menu (i.e. a sub-set of the CREATE command). The current value of any parameter can be preserved by simply pushing the return key at the prompt for defining that parameter. When more than a few parameters on a particular menu are to be changed, it can be most efficient to select ALL. Entering Z at any prompt stops the sequential prompting. Conversely, when only a few parameters on a menu are to be changed, then may be faster to select them individually; the user is presented with same menu after each change operation, and so the entire CHANGE + RECORD-NAME command does not have to be re-entered.

### 3.2.1 Hull Offsets and Sectional Data

The hull form is defined by either offset data, sectional data (i.e. beam, draft and area coefficient) or a combination of both, although only one type can be used at an individual station[4].

Offsets are defined as matching $Y(i,j)$ and $Z(i,j)$ pairs, where $Y$ is the horizontal offset (distance from hull centreline to hull surface), $Z$ is the vertical offset (distance from hull baseline to hull surface), index $i$ denotes the station, and index $j$ denotes the offset number at this station. To change a particular offset, the user first defines the station number, $i$, and offset number, $j$, and is then sequentially prompted for new $Y$ and $Z$ values. A response of null input to either or both offsets is permitted. Defining explicit values of $i$ and $j$ for each offset pair is tedious when more than a few offsets must be changed, and so the user may define ranges of values for $i$ or $j$, or both.

Valid ranges are; $n : m$,
$n$ : END,
and ALL.

where $n$ defines an initial value for the index ($i$ or $j$), $m$ defines a final value for the index, END indicates that all index values from $n$ to the highest (20 for station number and 12 for offset number) are used, and ALL indicates that all possible index values are used. Abbreviated input is recognized for both ALL and END, as demonstrated in the following example.

[4] the station numbering convention used in SHIPMO and MOPRE places station 0 at the forward perpendicular and station 20 at the after perpendicular.

```
MOPRE Command => ch off

Enter station number (or range) => 3:5

 Station 3 has 12 offsets.

Enter offset number (or range) => 5:e

 Y(3,5) =  5.780            ! current value
 Y(3,5) => 5.77             ! new value

 Z(3,5) =  6.263
 Z(3,5) =>                  ! push Return, preserve current value

 Y(3,6) =  6.900
 Y(3,6) =>                  ! push Return, preserve current value

 Z(3,6) =  8.693
 Z(3,6) =>                  ! push Return, preserve current value

 Y(3,7) =  8.240
 Y(3,7) =>                  ! push Return, preserve current value

 Z(3,7) =  12.963
 Z(3,7) => 11.963           ! new value

 & etc......
```

When initially defining $Y$, $Z$ data, it is possible to copy the offsets at the previous station to the current station by entering "-1" at the prompt for defining NOFF (number of offsets). This is most useful when the ship being defined has significant parallel midbody, but is also useful when the offsets are defined for constant values of $Z$ (i.e. at constant waterlines). In the latter case, the CHANGE command is later used to modify the $Y$ values and the $Z$ values are only entered once.

When a particular station is defined by sectional data, then only the station index, $i$, or range of stations, $i = n : m$, is required. The user is sequentially prompted to define input for the beam, draft and area coefficient, SHIPMO record (i2a).

### 3.3 CREATE and File Initialization

The CREATE command defines a new input file from scratch. After CREATE is entered, MOPRE initializes all parameters and prompts the user to provide input for all input records and related sub-records. This sequential prompting, called the CREATE mode, continues until either all input are defined or the user cancels the CREATE mode by entering Z at any prompt. The CREATE mode sequential prompting can be resumed at any time by entering CONTINUE_CREATING, as described in the next section.

The CREATE mode prompting is the same as used by the CHANGE command, except that the menus used by CHANGE are by-passed. Also, the CREATE mode does not accept null input (i.e. push Return) at prompts for defining numerical input. This is related to the file initialization procedure, in which every SHIPMO parameter is set to the initial, non-numeric value of "ZZZZ". Any parameter with this initial value cannot be by-passed by entering null input: the only acceptable input in the CREATE mode are; ?, Z, or a number within the range of acceptable values defined by the SHIPMO user's guide [5]. The only exceptions are for the title and parameters defining disk file names, for which any **non-Z** alpha-numeric input is acceptable.

When defining offset data, the user can copy the $Y$, $Z$ data of the previous station to the current station by entering "-1" at the prompt for defining NOFF, as described in the previous section.

### 3.3.1 Mixing CREATE and CHANGE Commands

One draw-back to the CREATE mode is that the user must provide values for all parameters, or cancel the CREATE mode. It is possible to restart the CREATE mode by entering CONTINUE_CREATING, but prompting always begins at the last undefined parameter. Thus, these commands cannot be used to by-pass a record or records for which input are not currently available; however, it is possible to by-pass records by combining the CREATE and CHANGE commands, as follows.

Suppose the user wants to produce a file to define hull offsets, but sea state data are not yet available. The CREATE mode is used to define the title and as many control integers as are known (e.g. system of units, water density, etc.), and then the CREATE mode is cancelled by entering Z. Next, the user enters the command CHANGE HULL FORM ALL, and is then sequentially prompted for all hull form data, including the offsets. It is not possible to immediately enter CHANGE OFFSETS, as some information, including the number of offsets at each station, must be defined first. After the offsets are input, the file is SAVED and can be ENTERED at a later time for completion.

## 3.4 CONTINUE_CREATING

The CONTINUE_CREATING command restarts the CREATE mode at the first undefined parameter. The record/sub-record in which this parameter is located is displayed for the user's information before the prompt for input. When a fragmented file is encountered (caused by mixing the CREATE and CHANGE modes as described above), then any parameters or records already defined are by-passed.

## 3.5 FIND and ENTER

The FIND command displays the title, file name and status (complete or incomplete) of library files. Library files are created automatically by the SAVE command. The user can define an optional "search string" as input to the FIND command, in which case only those files containing the search string in their titles are displayed. Note that the FIND command can only locate files that were previously SAVED by MOPRE. Additionally, if the multiple-directory structure described in Section 5.4 is used, then FIND only locates files in the current sub-directory.

The ENTER command loads a SHIPMO input file into MOPRE. ENTER requires the user to define the file name, which can be located using FIND. Note that any valid SHIPMO file can be ENTERED, provided its file name is known (i.e. ENTER is independent of FIND).

If the user attempts to ENTER a file while an altered (i.e. CHANGED but not SAVED) file is loaded in MOPRE, then a warning is issued and the ENTER command may be cancelled.

## 3.6 EXIT

The user may EXIT from any MOPRE prompt by entering EXIT, QUIT, BYE, or Z. If the user attempts to EXIT from MOPRE (i.e. stop the program) while an altered file is loaded, then a warning is issued and the EXIT command may be cancelled.

## 3.7 HELP

The HELP facility used in MOPRE is a module of the USERIO package which controls the access to and display of help messages which are stored in disk files. In MOPRE, the following HELP facility is provided.

```
MOPRE Command => he

Help is available for:

 READ_ME_FIRST        CLE        Batch      Change     Continue_creating
 Create     Define    Delete     Enter      Exit       Find       HELP
 Old-files  Save      Set        Show       Symbols

Topic ? => he
```

The MOPRE help messages displayed by this help facility are not described here, as they simply provide an on-line summary of the documentation contained in this Technical Communication.

## 3.8 SAVE

The SAVE command writes the current input in MOPRE to a disk file. The name of the disk file is defined by the user as a part of the SAVE command. When the SAVED file is complete (i.e. all input defined), then a copy is sent to the special SHIPMO input file, FOR002.DAT. Otherwise, only the user-named library file is written.

Note that a file cannot be saved until the title is defined.

## 3.9 SHOW

The SHOW command is used to display current values of SHIPMO input parameters and a variety of other items, as follows.

```
MOPRE Command => sh?

 Valid options are:

Bell                       Command_Dictionary         COMPLETE_RECORDS
Dictionary                 INCOMPLETE_RECORDS         RECORD
Symbols                    Time                       Verify

 SHOW ? =>
```

SHOW COMPLETE_RECORDS and its converse SHOW INCOMPLETE_RECORDS show the status of the current file and note which records and sub-records are affected. The SHOW RECORD command, followed by the name of a record or sub-record, displays the current values of the parameters in that record. As discussed previously, SHOW RECORD is the default comand in MOPRE, and so any record or sub-record name can be displayed by simply entering its name.

The SHOW sub-commands COMMAND DICTIONARY, DICTIONARY, SYMBOLS, and VERIFY commands support the CPARSE symbols described in Section 4.4. The SHOW BELL command simply shows the current bell status (bell can be SET ON or OFF), and the SHOW TIME command displays the current date and time.

When the user requests more information than can be displayed on one screen, then the program pauses when the screen is full and writes the following prompt.

```
 ** push Return to continue.....
```

After reading the on-screen text, the user can simply push the Return key to continue with the next screen or can enter Z to cancel subsequent screens. This is especially useful when using the SHOW command on a large amount of information (e.g. SHOW RECORD ALL displays all input, SHOW RECORD OFFSETS displays the entire table of offsets).

### 3.10 Other Commands; DEFINE, DELETE and SET

The DEFINE and DELETE commands are only used for the CPARSE symbols described in the next section. The SET command allows the user to set the terminal's warning bell on and off, and set the "verify mode" on and off. The verify mode is used to debug symbols, as discussed in the next section.

## 4 General Properties of User-Program Interface

The MOPRE user-program interface is controlled by the USERIO package [6], which provides abbreviation recognition, error detection and a variety of other features including a command-line editor and general-purpose HELP facility. The following USERIO modules are used by MOPRE.:

| | |
|---|---|
| VMSFOR | system-dependent FORTRAN calls for SHOW TIME command |
| READS | user-prompting, reading input, line parsing, menus and general-purpose character/number algorithms |
| CLE | Command-Line Editor |
| HELP | HELP facility |
| CPARSE | command and symbol parsing, and dictionary access/search algorithms |
| DIC2 | control algorithms for internal dictionaries |
| INTDIC | contains internal (BLOCK DATA) dictionaries |

The last two USERIO modules are an integral part of CPARSE but they are separated to facilitate creating and changing internal dictionaries. Each USERIO module is completely documented in the source code and the HELP facility provided with MOPRE contains detailed help on CLE and on the HELP facility itself. Also, the example/template program USER provided with the USERIO package provides complete on-line help for all aspects of USERIO operation.

The USERIO modules can be accessed by any FORTRAN program at a variety of levels, using procedures described in the source code. The remainder of this section describes the features provided by the USERIO package, as implemented in MOPRE.

### 4.1 Prompts

All requests for user-supplied input are preceeded by an informational prompt. For example, the top-level prompt used by MOPRE is

```
MOPRE Command =>
```

The information in the prompt changes according to the type of input expected. When MOPRE requests input to define a particular parameter, then the prompt includes a description of that parameter, its name (as defined in Reference 5) and when appropriate, its units. For example,

```
Seaway Spectrum Indicator (ISPEC) =>
```

where ISPEC is the SHIPMO parameter for defining which of three seaway spectral representations is used.

In cases where MOPRE asks the user to answer a yes or no question, then an assumed response is defined in the prompt within <...> brackets. This assumed response is used when the user simply pushes the Return key, as follows.

```
Are you sure you want to do this, Yes or No ?  <No> =>
```

## 4.2 Special Input: ?, Z, the Return key, !, and ∧

### On-Line Help, ?

At any prompt, it is possible to receive help by entering a question mark. The type of information given in response to ? depends on the context of the current command. In general, if no command is input or a command is not complete, then ? shows what commands or sub-commands are acceptable. When a command is complete, then ? describes the parameter or action indicated by the command. For example,

```
MOPRE Command => ch ?

 Valid options are:

TITLE                      CONTROL_INTEGERS        WAVE_FREQUENCIES
ENCOUNTER_FREQUENCIES      SEA_STATE               SHIP_SPEEDS
HULL_FORM                  BASIC_SHIP_DATA         STATION_DEFINITION
OFFSETS                    COEFFICIENTS            LOAD_WATERLINE
SEAKEEPING_AND_SLAMMING    APPENDAGES              BILGE_KEELS
SKEG                       FINS                    SHAFT_BRACKETS
RUDDER                     ANTI-ROLLING_TANK

 CHANGE ? => con
```

```
        CONTROL INTEGERS
        ----------------
     1  Input units (IN)
     2  Output units (IOUT)
     3  Water density (IRHO)
     4  Seaway spectrum (ISPEC)
     5  Wave train corrections (ICORR)
     6  Regular wave response (IRESP)
     7  Disk file indicator (IPPF)
     8  ALL

 Enter selection (integer) => 4

Seaway spectrum indicator (ISPEC) => ?

      0  Quadratic regression spectrum
      1  Bretschneider two parameter spectrum
      2  Measured spectrum

Seaway spectrum indicator (ISPEC) =>
```

Note that if ? is entered and no additional information is provided (i.e. same prompt reappears immediately), then information may be available from the HELP facility, described earlier.

### Exit, Z

At any prompt, it is possible to exit by entering Z, EXIT, QUIT, or BYE. In most cases, after the exit-word is input, the program execution passes to the previous level. Input of an exit-word at the top-level of MOPRE stops program execution. If MOPRE contains a file that has been changed since the last SAVE command was issued, then the user is warned and given an opportunity to cancel the exit command.

### The Return key

Aside from its function in entering typed commands, the Return key is used for a variety of other purposes. The phrase "null input" describes the condition when the Return key is pushed with no preceeding input. Note that for most purposes, the Enter key found on some keyboards and most numeric keypads is identical to the Return key.

The response to null input varies with the context of the current command. When the CHANGE command is completed and the user is prompted for numerical input, then null input indicates that the user does not want to change the current value. In this case, one of two actions will follow the null input: if the parameter has a current value or a pre-set default value, then program execution continues and the parameter is not affected; otherwise, the terminal will beep and the user is reprompted for the same input. This last event indicates that the user must supply input for the parameter; however, input of Z or ? will produce the results described earlier.

When a command is only partially complete, then null input cancels the command and the program returns to the top-level command prompt. This cancellation occurs at the second step of the command recognition process. In the first step, the user types a command and enters it by pushing the Return key. This command is then interpreted by the program and if it is incomplete, the user is prompted for more information. Null input at this second prompt cancels the command, as follows.

```
MOPRE Command => ch

CHANGE ? =>        ! push Return_key

MOPRE Command =>
```

When the user enters null input at the prompt for a menu, then one of two actions may follow: if there are more items for the menu (i.e. too many to display on one screen), then the remainder are displayed; otherwise, the null input is equivalent to Z, and the program returns to the next higher level. If the same menu reappears after null input, then the program requires the user to select an item from the menu (i.e. exit from this menu is not possible).

### Invoke Command-Line Editor, ∧

The special character ∧ can be typed at any prompt to invoke the Command-Line Editor, CLE, which retrieves the most recent command and presents the following display.

```
MOPRE Command => ^

CLE: Command_Line Editor
------------------------

Command: This is the most recent command
 Cursor: ^
    CLE:
```

At the "CLE:" prompt, the user may enter ? for help on CLE, push the Return key (i.e. null input) to process the current command, or use the CLE movement and editing commands to change this or any of the last 40 commands. Alternately, the user may quit and return to the top-level command prompt by entering Z.

When using a recognized video terminal, cursor movement and editing functions (find, replace, delete, insert and overwrite) are performed without screen scrolling. Otherwise, each movement or editing command produces a new group of Command:, Cursor: and CLE: information lines which reflect the requested change. Recognized terminals and a summary of CLE commands are available from within CLE in response to the ? input. Details of all CLE commands and procedures are available from the HELP facility, described earlier.

### Comment Line, !

Any line or part of a line preceeded by an exclamation mark, !, is considered to be a comment and is completely ignored. If valid commands or parts of commands preceed the !, they are interpreted normally.

## 4.3 Error Detection and Resolution

Three types of errors are detected: ambiguous input; incomplete input; and, unrecognized input. When an error is encountered in multiple-word input, then only the incorrect parts are affected.

Ambiguous input is resolved by displaying the possible matches in a simple menu and prompting the user to select the desired match. If none of the choices are acceptable, then null input cancels the command.

The response to incomplete input has already been demonstrated in the section discussing response to the Return key. In general, after the user pushes the Return key to enter an incomplete command, the command is expanded to its full form and the user is prompted for more input. At this prompt, the user can enter ? for a display of acceptable sub-commands, push the Return key to cancel the command or enter an appropriate sub-command.

Unrecognized input usually indicates a typographical error. The unrecognized command or sub-command is displayed and the user is prompted for a correction.

The methods used to resolve these three types of errors are illustrated by the following example in which the user attempts to enter the command SHOW RECORD SEA STATE.

```
MOPRE Command => sk rec

? Do not recognize SK

? => sh

SHOW RECORD ? => s

? Ambiguous
```

```
  1 SEA_STATE
  2 SHIP_SPEEDS
  3 STATION_DEFINITION
  4 SEAKEEPING_AND_SLAMMING
  5 SHAFT_BRACKETS
  6 SKEG

 Enter selection (integer) => 1
```

At the end of this dialogue, the command SHOW RECORD SEA_STATE is complete and the program responds by showing the current values of the sea state parameters.

## 4.4 Symbols

A symbol is a customized abbreviation for a command which is intercepted and interpreted by the command parser, CPARSE. Consider a symbol with user-defined name T: and user-defined definition SHOW TIME. Note that all symbol names must end with a colon (:).

```
MOPRE Command => t:

It is now 4-MAR-1988 14:00:18.52

MOPRE Command =>
```

Symbols offer two benefits, they are easy to type (provided the user defines a suitably short name), and they are processed very quickly. The rapid processing time is due to the way symbols are handled by CPARSE. The symbol is replaced by its definition and immediately sent to the application program (MOPRE in this case), by-passing the error detection and command completion algorithms normally used. This means that a symbol definition cannot contain abbreviations (e.g. a symbol with name T: and definition SHO TIM would not be recognized).

Once defined (command DEFINE SYMBOL), a symbol is available for immediate use in any subsequent execution of the program. The commands SHOW SYMBOLS, DEFINE SYMBOL and DELETE SYMBOL are available in MOPRE. Also, the commands SHOW COMMAND_DICTIONARY, SHOW DICTIONARY, SET VERIFY ON or OFF, and SHOW VERIFY support symbol definition: the two dictionary-related commands are used to determine the correct syntax for multiple-word symbols; and, the VERIFY mode provides a simple debugging tool, illustrated below. Dictionaries are explained in detail under HELP SYMBOLS in the on-line MOPRE help facility.

More advanced symbol definitions are described in the example/template program USER provided with the USERIO package and in the MOPRE HELP facility; however, the restricted command-set used in MOPRE limits their application.

### 4.5 Verify Mode

As mentioned in the previous section on MOPRE commands, the SET VERIFY ON or OFF comand is used to debug symbols. Since the user-supplied symbol definition is not verified by CPARSE and is not normally shown on-screen, it can be difficult to determine why a symbol does not work correctly. When the SET VERIFY ON command is used, then the actual command received by MOPRE is displayed on-screen. For example, suppose the symbol T: has an incorrect definition of "SHOE TIME".

```
MOPRE Command => t:

** Unrecognized Command

MOPRE Command => se ver on

MOPRE Command => t:

VERIFY ON, command received by MOPRE is:

  SHOE TIME

** Unrecognized Command
```

When an error is detected in a symbol definition, it is possible to correct the error using CLE. This is done by entering DEFINE SYMBOL, entering the name of the symbol to be edited at the prompt for a new symbol name, and then invoking CLE by entering the ∧ symbol. Alternately, the original definition can be replaced by simply entering a new one.

## 5 Computer Requirements

### 5.1 FORTRAN

MOPRE and the USERIO user-interface algorithms used by MOPRE conform to the FORTRAN-77 standard (ANSI X3.9-1978) with only two exceptions:

1. the non-standard VAX VMS FORTRAN symbol '$' is used to suppress the automatic carriage return at the end of a formatted WRITE statement, and

2. the non-standard VAX VMS FORTRAN call statements CALL DATE and CALL TIME are used for the MOPRE command SHOW TIME.

The '$' format symbol is used in both MOPRE and USERIO. Other computers may not use the same symbol, but should have the same capability. Similarly, the operating system calls used for SHOW TIME should be available on most computers, although the details of their usage may change. These call statements are isolated in the source VMSFOR, and so they can be modified without affecting the MOPRE source.

## 5.2 LINK Requirements

On a VAX VMS operating system, the executable version of program MOPRE is produced by the following statement.

```
$ LINK MOPRE, [USERIO]VMSFOR, READS, CLE, HELP, CPARSE, DIC2, INTDIC
```

MOPRE is the main program and contains the code for performing all commands except for the user-program interface functions performed by the USERIO package, as described in Section 4. The [USERIO] directory specifier in this LINK command defines that the USERIO module VMSFOR and all following modules (i.e. files) are located in a separate directory.

## 5.3 Command Dictionaries

The MOPRE command structure described in preceeding sections is defined for the USERIO command-parser CPARSE in disk files called command dictionaries. In general, each line in a command dictionary defines the name of a command, and any requirements for sub-commands (in different command dictionaries) or other input. The external (i.e. disk file) command dictionary system used by MOPRE can be converted to an internal (BLOCK DATA) system by following the directions given in the INTDIC source code. The command dictionaries used by MOPRE are listed below, but their contents are not described as they are self-evident on examination.

The MOPRE command dictionaries are, in alphabetical order;

| | |
|---|---|
| APPDK.DAT | names of appendage sub-records |
| DEFDIC.DAT | sub-commands for DEFINE |
| DELDIC.DAT | sub-commands for DELETE |
| HULLDK.DAT | names of hull form sub-records |
| MODIC.DAT | top-level command dictionary |
| RECDIC.DAT | names of input records |
| SETDIC.DAT | sub-commands for SET |
| SHODIC.DAT | sub-commands for SHOW |

| | |
|---|---|
| SYMBDK.DAT | stores user-defined symbol names and definitions |
| TOGDIC.DAT | auxiliary dictionary for SET, contains ON and OFF |

Dictionary MODIC.DAT is the top-level command dictionary, which defines the command verbs (e.g. CHANGE, SAVE, etc.) and establishes connections to other dictionaries. The on-line help provided with the example/template program USER and comments in CPARSE source code document all aspects of dictionary usage.

## 5.4 VMS Environment

The following menu-system for controlling program execution and other aspects of running MOPRE and SHIPMO is available for the VAX VMS operating system.

```
 SEAKEEPING: Program Control
 --------------------------
  1  Run MOPRE
  2  Run SHIPMO
  3  Run PHHS7 (pitch and heave in head seas)
  4  Run MSPA (monohull seakeeping performance assessment)
  5  Run TISIM (real-time simulation)
  6  Submit Batch Job
 96  Lineprinter Control
 97  Edit
 98  Help
 99  EXIT

Enter selection (integer) =>
```

The source for this menu-system is written in Digital Command Language, DCL [8], and is provided with the MOPRE source. It is important to appreciate that this menu-system is completely independent of the FORTRAN programs MOPRE and SHIPMO, and that it is not required for their operation; however, it does provide a convenient environment.

The menu-system shown here is implemented on the DREA Hydronautics VAX computer, and so it includes the seakeeping analysis programs PHHS7 [9, 10 and 11], MSPA [12] and TISIM [13]. Menu item 6 is used to submit a batch job for SHIPMO, controlled by the command file produced by the BATCH command in MOPRE. Menu item 96 allows the user to send the SHIPMO lineprinter output to a disk file where it can be examined using a system editor (menu item 97).

The functions provided by this menu-system can be easily duplicated on most computers. The only aspect of its operation considered here is its multiple-directory structure.

### Directories

The MOPRE/SHIPMO package is implemented on the DREA Hydronautics VAX computer using the following directories.

[SHIPMO] main directory: controls login operation and contains executable version of all FORTRAN programs, master file for defining and locating MOPRE command dictionaries, and DCL procedure for program-control menu-system

[SHIPMO.HELP] contains help files used by MOPRE HELP facility

[SHIPMO.DIC] contains MOPRE command dictionaries

[SHIPMO.SOURCES] contains source versions of all FORTRAN programs

[SHIPMO.USERx] one of any number of user sub-directories which contain the SHIPMO input files produced by that person;
sub-directory selection controlled by login procedure,
each user sub-directory has its own MOPRE-controlled library.

The contents of file MASTER.DAT, located in directory [SHIPMO] defines the location of all command dictionaries (i.e. [SHIPMO.DIC] for this version of MOPRE). The top-level help file HLPTXT.DAT, located in directory [SHIPMO.HELP] includes references to other help files located in different directories. The MOPRE code defines the location of HLPTXT.DAT for the HELP module. The HELP source code and on-line help provided with MOPRE describe how references to other help files are defined and how to change them.

## 6 Concluding Remarks

The FORTRAN program MOPRE is an interactive, user-friendly pre-processor for creating, modifying and validating input files for the seakeeping analysis program SHIPMO.

An overview of MOPRE introduced the commands and general procedures for producing SHIPMO input files and managing library files. The MOPRE commands were described in detail with emphasis on procedures which simplify the definition and modification of ship geometry. The general features of the MOPRE interface provided by the USERIO package were described, including: on-line help, error detection, the Command-Line Editor and user-definable command abbreviations (symbols). Finally, the general computer requirements for MOPRE were discussed and an optional menu-system for program control was described.

# Appendix A: Using MOPRE to Create a New SHIPMO Input File

This appendix contains a transcript of the example terminal session described in the overview of MOPRE, in which the user creates a new SHIPMO input file. All user-supplied input to MOPRE is preceeded by a “=>” prompt and all editorial comments are enclosed within “---<comments>--- brackets.

The “SEAKEEPING: Program Control” menu shown below is the VAX/VMS Environment described in Section 5.4.

```
 SEAKEEPING: Program Control
 ---------------------------
 1  Run MOPRE
 2  Run SHIPMO
 3  Run PHHS7 (pitch and heave in head seas)
 4  Run MSPA (monohull seakeeping performance assessment)
 5  Run TISIM (real-time simulation)
 6  Submit Batch Job
96  Lineprinter Control
97  Edit
98  Help
99  EXIT

 Enter NUMBER => 1


MOPRE, Pre-processor for SHIPMO
------------------------------
** Initializing library file...    ---<10 sec>---

MOPRE Command => cre               ---<i.e. CREATE>---

 TITLE:  maximum 50 characters.
 -----
 TITLE => frigate example (see Appendix A, Reference 5)
 CONTROL INTEGERS :
 -----------------
Input units (IN) => ?

    0  British units
    1  Metric units

Input units (IN) => 0
Output units (IOUT) => 1
Water type or density (IRHO) => ?

    0  Salt water
    1  Fresh water

Water type or density (IRHO) => 0
Seaway spectrum indicator (ISPEC) => 22

 ---<beep>---

 ** Invalid Input: enter I, where 0 .LE. I .LE. 2

Seaway spectrum indicator (ISPEC) => 2
Wave train correction (ICORR) => 2
Wave response output (IRESP) => 2
Disk file storage (IPPF) => 0

 WAVE FREQUENCY DATA :
 --------------------
Lowest wave frequency (WMIN) [rad/sec] => .2
Highest wave frequency (WMAX) [rad/sec]=> 2
Frequency increment (DW) => .1

 ENCOUNTER FREQUENCY DATA :
 -------------------------
Lowest encount. freq. (WEMIN) [rad/sec]=> .1
Highest encount. freq. (WEMAX) [rad/sec]=> 5.9
Enter frequency increment (DWE) => .2
Hull sectional calc. (METHOD) => 0
2-D section data storage (ISAVE) => 0
```

```
 SEA STATE DATA :
 ----------------
No. of sea directions (NSD) => 0
HSW(1) => 10.66273
TSW(1) => 10
No. of spectral density freq. (ITOP) => ?

 The number of frequencies at which spectral density is
 being input (max. = 300)

No. of spectral density freq. (ITOP) => 73
Units for spectral density (ISEAIN) => 0
Format of spectral density (INDATA) => 1
Enter filename (SPECTR) => bretsc

 SHIP SPEED DATA :
 -----------------
Lowest ship speed (UKMIN) [knots] => 18
Highest ship speed (UKMAX) [knots] => 18
Ship speed increment (DUK) => 0

 BASIC SHIP DATA :
 -----------------
 Do you want variable or fixed waterline ?

If variable waterline is selected you should enter offsets up to
the deck edge and if fixed waterline is selected you must enter
offsets only up to the waterline.

     WATERLINE
     ---------
   1  Variable Waterline
   2  Fixed Waterline

 Enter selection (integer) => 1

 Variable waterline requires that at least one station
 has more than 8 offsets. The waterline will be calculated
 automatically.
```

```
Length between perpendiculars (EL) => 356
Height of CG (HCG) => 18.09
Metacentric height (GM) => 2.09
Waterplane z-coordinate (ZWL) => 0

 Select one of the following:

     RNF or RRGB
     -----------
   1  RNF Roll natural frequency
   2  RRGB Roll radius of gyration

 Enter selection (integer) => 2

Roll radius of gyration (RRGB) => .35
Yaw radius of gyration (YRGL) => .25
Roll radius control integer (IRG) => 0

 STATION DEFINITION :
 --------------------
Beam-wise scaling factor (BMSF) => 1
Draft-wise scaling factor (DTSF) => 1

 The current station is :  0

NOFF(0) => 0

    ---<beep>---

 *** WARNING ***
 IVWAT = 1, which signifies variable waterline. SHIPMO
 will not calculate a waterline with BEAM, DRAFT and
 ACOEF, so you must give the offsets for each station.
 This corresponds to an NOFF value > 1.

 The current station is :  0

NOFF(0) => ?
```

```
 NOFF(I) is the number of offsets for station I.
 (max. = 12) If NOFF(I) = 0  or 1 then the beam, draft,
 and area coefficient are used for that station.

 For station 0 and station 20 hitting return will set
 the zero defaults for that station.

 Entering -1 will copy the same data as the previous
 station for the current station.

 The current station is :  0

NOFF(0) =>             ---<push Return>---

 The current station is :  1

NOFF(1) => 12

YA(1,1) => 0
ZA(1,1) => 0

YA(1,2) => .46
ZA(1,2) => 0

YA(1,3) => 1.19
ZA(1,3) => 2.77

YA(1,4) => 1.82
ZA(1,4) => 6.6

YA(1,5) => 2.33
ZA(1,5) => 9.93

YA(1,6) => 2.57
ZA(1,6) => 11.21

YA(1,7) => 2.84
ZA(1,7) => 12.61

YA(1,8) => 3.5
ZA(1,8) => 15.53
```

```
YA(1,9) => 4.5
ZA(1,9) => 18.84

YA(1,10) => 6.55
ZA(1,10) => 23.84

YA(1,11) => 9.25
ZA(1,11) => 28.78

YA(1,12) => 12.79
ZA(1,12) => 34.01

IEDDY(1) => 1

 The current station is :  2

NOFF(2) => 12

   ---<data-entry prompts and input for
       remaining 19 stations are omitted>---

 LOAD WATERLINE DATA :
 --------------------
Control integer for loading (IBAL) => 1
Draft at midships (DMID) => 13.69
Trim by the stern (TRIM) => .5

 SEAKEEPING and SLAMMING :
 ------------------------
Number of seakeeping positions (NPOS) => 1
Time period for slamming calc. (THR) => 20
Exceedance parameter (SLAMEX) => .01

XST(1) => 3
ZPOS(1) => 11.16
FREEB(1) => 33.29
ISLAM(1) => 1
DEADR(1) => 34.5
HWFB(1) => .5
FFACT(1) => 0
```

```
 BILGE KEEL PAIRS :
 ------------------
Number of bilge keel pairs (NBKP) => 1

NFBK(1) => 10
NLBK(1) => 14

YBK(1,10) => 18.96
ZBK(1,10) => 7.01
BKK(1,10) => 2
ELBK(1,10) => 12.1

YBK(1,11) => 18.78
ZBK(1,11) => 6.1
BKK(1,11) => 2
ELBK(1,11) => 17.8

YBK(1,12) => 18.47
ZBK(1,12) => 6
BKK(1,12) => 2
ELBK(1,12) => 17.8

YBK(1,13) => 18.25
ZBK(1,13) => 6.38
BKK(1,13) => 2
ELBK(1,13) => 17.8

YBK(1,14) => 18.21
ZBK(1,14) => 7.13
BKK(1,14) => 2
ELBK(1,14) => 21.9

 SKEG :
 ------
Skeg aftermost station (XSK) => 17.87
Skeg breadth (BSK) => 3.9
Skeg length (ELSK) => 60
```

```
 FIN PAIRS :
 -----------
 Coordinates are input for the port foil; it is assumed
 that a counterpart exists on the starboard side. Note
 that y- and z- coordinates apply to the fin root.
 See the figure in the SHIPMO Users Manual for more info.

Number of fin pairs or U-tank (NFP) => 0

 SHAFT BRACKETS :
 ----------------

 A shaft bracket consists of two arms. For NSH  = 1, input
 data only for the port arm of the bracket. For NSH > 1,
 input data for both arms of each port bracket with the
 outboard arm preceeding the inboard arm.

Number of shaft brackets (NSH) => 2

X(1) => 19.3
Y(1) => 3.8
Z(1) => 9.7
B(1) => 7.35
CR(1) => 2.625
CE(1) => 2.625
CLAO(1) => 0
GAM(1) => 65
VEFF(1) => 1

X(2) => 19.3
Y(2) => 10.2
Z(2) => 10.5
B(2) => 7.4
CR(2) => 2.625
CE(2) => 2.625
CLAO(2) => 0
GAM(2) => 79
VEFF(2) => 1
```

```
 RUDDER :
 --------
Number of rudders (NRD) => 2
Station for rudder (X) => 19.16
Horizontal coordinate for twin rudders (Y) => 6.5
Vertical coordinate for rudder (Z) => 5.85
Span of rudder (B) => 9.92
Root chord of rudder (CR) => 7.5
Tip chord of rudder (CE) => 6.017
Lift curve slope of rudder (CLAO) => 0
Propeller slipstream indicator (GAM) => 1
Correction factor (VEFF) => 1

 Rudder Roll Gains, Natural Freq. and Damping :
 ---------------------------------------------
 The variables QFDDR, QFDR, QFR, WLR and WHR are used to
 simulate a rudder roll stabilization system. If you do
 not wish to consider this input 0 for these variables.

Rudder roll acceleration gain (sec*sec) (QFDDR)=> 0
Rudder roll velocity gain (sec)(QFDR) => 0
Rudder roll gain (QFR) => 0
Rudder control system natural frequency (rad/sec)(WR) => 0
Rudder control system damping ratio (ZETR) => 0
Low frequency cutoff for rudder roll stabilizer (rad/sec)(WLR) => 0
High frequency cutoff for rudder roll stabilizer (rad/sec)(WHR) => 0

 Rudder Yaw Gains :
 -----------------
 These are autopilot inputs which may be used to simulate
 a steering system. If a model of the steering system is
 not required, you are advised to input zero for all of
 QYDDR, QYDR, and QYR.

Rudder yaw acceleration gain (sec*sec)(QYDDR) => 0
Rudder yaw velocity gain (sec)(QYDR) => 0
Rudder yaw gain (QYR) => 0

** FILE CREATED, enter ? for HELP
```

```
MOPRE Command => ?          ---<i.e. help>---

 Commands are:

BATCH         CHANGE        CONTINUE_CREATING           CREATE        DEFINE
DELETE        ENTER         EXIT            FIND        HELP          SAVE
SET           SHOW

 or.....a record or subrecord name (SHOW RECORD ? for listing)

 Enter Command => sa          ---<i.e. SAVE>---

 SAVE (filename) ? => example

 ** Saving current file..., Please wait **

MOPRE Command => z            ---<i.e. EXIT>---


 SEAKEEPING: Program Control
 ---------------------------
 1  Run MOPRE
 2  Run SHIPMO
 3  Run PHHS7 (pitch and heave in head seas)
 4  Run MSPA (monohull seakeeping performance assessment)
 5  Run TISIM (real-time simulation)
 6  Submit Batch Job
96  Lineprinter Control
97  Edit
98  Help
99  EXIT

 Enter NUMBER => 99


      ---<end of Appendix A>---
```

## Appendix B: Using the CHANGE Command to Modify an Existing File

This appendix shows the second example terminal session described in the overview of MOPRE, in which an existing file is modified using the CHANGE command.

```
 MOPRE, Pre-processor for SHIPMO
 -------------------------------
 MOPRE Command => find frig       ---<i.e. FIND a file with 'frig'
                                           in its title>---
 FILE NAME: SHIPM4
     TITLE: FRIGATE EXAMPLE
    STATUS: COMPLETE FOR SHIPMO

 FILE NAME: EXAMPLE
     TITLE: FRIGATE EXAMPLE (SEE APPENDIX A REFERENCE 5)
    STATUS: COMPLETE FOR SHIPMO

MOPRE Command => en example      ---<i.e. ENTER file called EXAMPLE>---

 ** Reading in file.....

MOPRE Command => ch tit          ---<i.e. CHANGE TITLE>---

TITLE =  frigate example (see Appendix A, Reference 5)
TITLE => frigate example 2

MOPRE Command => contr           ---<i.e. SHOW RECORD
                                          CONTROL_INTEGERS>---
 CONTROL INTEGERS
 ----------------
 IN     = 0
 IOUT   = 1
 IRHO   = 0
 ISPEC  = 2
 ICORR  = 2
 IRESP  = 2
 IPPF   = 0

MOPRE Command => ch cont         ---<i.e. CHANGE CONTROL_INTEGERS>---
```

```
 CONTROL INTEGERS
 ----------------
 1  Input units (IN)
 2  Output units (IOUT)
 3  Water density (IRHO)
 4  Seaway spectrum (ISPEC)
 5  Wave train corrections (ICORR)
 6  Regular wave response (IRESP)
 7  Disk file indicator (IPPF)
 8  ALL

Enter selection (integer) => 4

Seaway spectrum indicator (ISPEC) =  2      ---<current value>---
Seaway spectrum indicator (ISPEC) => 1

** NOTE: Major changes have been made to the SEA_STATE record.
         Current default values may have to be changed.

** Push Return to continue....

SEA STATE DATA :
----------------
NSD    = 0
ANGLE  = 0.0
NSEA =  1
HSW(1)  = 10.66273            TSW(1)  = 10

 Do you want to change any/all of these parameters?

 Change SEA_STATE, Yes or No  <Yes>  =>        ---<push Return>---

 SEA STATE
 ---------
 1  Number of sea directions (NSD)
 2  Spreading angle (ANGLE)
 3  Principal sea directions (PSDIR)
 4  Number of seaways (NSEA)
 5  Significant wave height and period(HSW and TSW)
 6  ALL

Enter selection (integer) => 6
```

```
No. of sea directions (NSD) =  0
No. of sea directions (NSD) => 2

 Since NSD no longer 0 you must give a value for ANGLE
 Spreading angle (ANGLE) => 60

Principal Sea Directions: PSDIR(1) => 180
                          PSDIR(2) => 90

Number of seaways (NSEA) =  1
Number of seaways (NSEA) => 2

Do you want to re-enter all the values for HSW and TSW;
otherwise just enter the additional values.

Enter all values ? Yes or No  <No> =>         ---<push Return>---

HSW(2) => 15
TSW(2) => 13

   SEA STATE
   ---------
   1  Number of sea directions (NSD)
   2  Spreading angle (ANGLE)
   3  Principal sea directions (PSDIR)
   4  Number of seaways (NSEA)
   5  Significant wave height and period(HSW and TSW)
   6  ALL

 Enter selection (integer) =>                     ---<push Return>---

MOPRE Command => sea_           ---<i.e. SHOW RECORD SEA_STATE>---

 SEA STATE
 ---------
 NSD    = 2
 ANGLE  = 60
 Principal sea directions: PSDIR(1) = 180
                           PSDIR(2) = 90
 NSEA = 2
 HSW(1)  = 10.66273          TSW(1) = 10
 HSW(2)  = 15                TSW(2) = 13
```

```
MOPRE Command => ch off                    ---<i.e. CHANGE OFFSETS>---

 Since a station can have either coefficients or offsets
 all the stations can be accessed and changed together.
 To convert a station to offsets from coeff. or vice versa
 you must first change NOFF(I) using the command
 CHANGE STATION_DEFINITION

Enter station no. (or range) => ?        ---<i.e. eh, what ?>---

 Acceptable responses are of the form i:j, ALL, i:END
 where i and j are integers between 0 and 20.
 Enter Z to exit

Enter station no. (or range) => z

MOPRE Command => off                 ---<i.e. SHOW RECORD OFFSETS>---

 STATION DEFINITION, COEFFICIENTS, and OFFSETS :
 -----------------------------------------------
 BMSF   = 1
 DTSF   = 1
 NSTOT  = 21
 NST    = 21

 Station:  0   XA(0) = 0    No. of Offsets = 0    IEDDY = 1
 Coeff:        Beam = 0.0   Draft = 0.0           Area Coeff. = 0.0

 Station :  1   XA(1) = 1    No. of Offsets = 12   IEDDY = 1

  Y -   0.000   0.460   1.190   1.820   2.330   2.570   2.840   3.500
  Z -   0.000   0.000   2.770   6.600   9.930  11.210  12.610  15.530

  Y -   4.500   6.550   9.250  12.790
  Z -  18.840  23.840  28.780  34.010


 ** push Return to continue......z       ---<NOTE: Z cancels display
                                             of remaining stations>---
```

```
MOPRE Command => ch off

 Since a station can have either coefficients or offsets
 all the stations can be accessed and changed together.
 To convert a station to offsets from coeff. or vice versa
 you must first change NOFF(I) using the command
 CHANGE STATION_DEFINITION

Enter station no. (or range) => 1

 For station  1 there are 12 offsets

Enter offset (or range) => 3:5

YA(1,3) =  1.19
YA(1,3) =>            ---<push Return = preserve current value>---

ZA(1,3) =  2.77
ZA(1,3) => 2.85

YA(1,4) =  1.82
YA(1,4) =>

ZA(1,4) =  6.6
ZA(1,4) =>

YA(1,5) =  2.33
YA(1,5) =>

ZA(1,5) =  9.93
ZA(1,5) => 10.2

Enter station no. (or range) => z

MOPRE Command => sav example

 This SAVE will overwrite the file: EXAMPLE
 Do you want to continue this process ?

 Continue and overwrite, Yes or No ?  <Yes> => n

 ** SAVE Cancelled.
```

```
MOPRE Command => save exampl2

 ** Saving current file.....

MOPRE Command => z


 SEAKEEPING: Program Control
 ---------------------------
  1  Run MOPRE
  2  Run SHIPMO
  3  Run PHHS7 (pitch and heave in head seas)
  4  Run MSPA (monohull seakeeping performance assessment)
  5  Run TISIM (real-time simulation)
  6  Submit Batch Job
 96  Lineprinter Control
 97  Edit
 98  Help
 99  EXIT

 Enter NUMBER => 99

      ---<End of Appendix B>---
```

# REFERENCES

1. Schmitke, R.T. and Whitten, B.W.: "SHIPMO - A FORTRAN Program to Predict Ship Motions in Waves", DREA Technical Memorandum 81/C, October 1981.

2. Ando, S.: "Wave Load Prediction for the SHIPMO Computer Program" DREA Technical Memorandum 82/L, October 1982.

3. Graham, R. and Millar, G.: "SHIPMO2: An Updated User's Manual for the SHIPMO Computer Program Incorporating Measured Sea Spectra and Wave Loads", DREA Technical Memorandum 84/G, May 1984.

4. Graham, R.: "SHIPMO3: Improved Viscous Roll Damping Predictions for the SHIPMO Computer Program", DREA Technical Memorandum 86/212, May 1986.

5. Graham, G. and Trudelle, C.: "SHIPMO4: An Updated User's Manual for the SHIPMO Computer Program Incorporating an Extended Hydrostatics Capability and an Improved Viscous Roll Damping Model", DREA Technical Communication 87/304, March 1987.

6. Colwell, J.L.: "USERIO: FORTRAN User-Program Interfaces and General Purpose Algorithms for Character and Number Manipulation", DREA Technical Communication, in preparation.

7. "General Information", VAX/VMS (version 4.3), Volume 2, Digital Equipment Corporation, Maynard, Mass., July 1985.

8. "DCL Dictionary", VAX/VMS (version 4.3), Volume 2, Digital Equipment Corporation, Maynard, Mass., July 1985.

9. Mackay, M. and Schmitke, R.T.: "PHHS - A FORTRAN Programme for Ship Pitch, Heave and Seakeeping Prediction", DREA Technical Memorandum 78/B, April 1978.

10. Schmitke, R.T.: "Improved Slamming Predictions for the PHHS Computer Program", DREA Technical Memorandum 79/A, February 1979.

11. Schmitke, R.T. and Colwell, J.L.: "Addition of an Experimental Data Base to the PHHS Computer Program", DREA Technical Memorandum 82/H, June 1982.

12. Schmitke, R.T.: "MSPA - A Computer Program Package for Seakeeping Performance Assessment of Monohull Ships", DREA Technical Memorandum 83/A, January 1983.

13. Graham, R.: "TISIM - A Real Time Computer Simulation of Ship Motions in Irregular Seas", DREA Technical Memoramdum 84/AA, December 1984.

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM
(highest classification of Title, Abstract, Keywords)

## DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1 ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

Defence Research Establishment Atlantic

2 SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable)

UNCLASSIFIED

3 TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C,R or U) in parentheses after the title.)

MOPRE: A Pre-Processor for Defining and Modifying Input to the Seakeeping Analysis Program SHIPMO

4 AUTHORS (Last name, first name, middle initial. If military, show rank, e.g. Doe, Maj. John E.)

Colwell, J.L., Colter, D.J., Lamanque, F.

5 DATE OF PUBLICATION (month and year of publication of document)

OCTOBER 1988

6a NO OF PAGES (total *containing information Include* Annexes, Appendices, etc.)

42

6b NO OF REFS (total cited in document)

13

6 DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Technical Communication

8 SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

9a PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. *Please specify whether project or grant*)

1AH-13

9b CONTRACT NO (if appropriate, the applicable number under which the document was written)

10a ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

DREA TECH COMMUNICATION 88/303

10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11 DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12 DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availabilty (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM

13. **ABSTRACT** ( a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both offical languages unless the text is bilingual).

The FORTRAN program MOPRE is an interactive, user-friendly pre-processor for creating, modifying and validating input files for the 'strip-theory' seakeeping analysis program SHIPMO. An overview introduces the commands and general procedures used in MOPRE to produce the SHIPMO input and defines the record/sub-record structure used by both programs. The MOPRE commands are described in detail, with emphasis on procedures which simplify defining and modifying ship geometry. The general features of the MOPRE interface are described, including: on-line help, abbreviation recognition, error detection, and a Command-Line Editor. Computer requirements are discussed and an optional, operating system-dependent menu-system for controlling execution of MOPRE, SHIPMO and other seakeeping programs is described.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

SHIPMO
Seakeeping
FORTRAN
Pre-Processor
Ship Motions
Ship Motion Computer Program